2019

Marginal maximum likelihood estimation of conditional autoregressive models with missing data

Thomas F. Suesse
University of Wollongong, tsuesse@uow.edu.au

Andrew Zammit-Mangion
University of Wollongong, azm@uow.edu.au

Marginal maximum likelihood estimation of conditional autoregressive models with missing data

Abstract

Maximum likelihood (ML) estimation of spatial autocorrelation models is well established for the case where each node in the graph is directly observed. When one or more nodes are not observed, the user has a variety of computational tools at her or his disposal ranging from the expectation-maximization algorithm, which has become a standard for missing-data problems, to marginal likelihood estimation methods and to fully Bayesian approaches. In this article, we give a comprehensive overview of likelihood-based computational frameworks for parameter estimation of the conditional autoregressive model, and we establish connections with several algorithms in the literature that are iterative and often computationally suboptimal. We show that a vanilla marginal ML approach, which we provide computational details for, is still generally orders of magnitude faster than the iterative approaches, even on large data sets and especially so when the number of unobserved units is relatively large.

Disciplines

Engineering | Science and Technology Studies

Publication Details

Suesse, T. & Zammit-Mangion, A. (2019). Marginal maximum likelihood estimation of conditional autoregressive models with missing data. Stat, 8 (1), e226-1-e226-10.

Marginal Maximum Likelihood Estimation of Conditional Autoregressive Models with Missing Data

Thomas Suesse and Andrew Zammit-Mangion

School of Mathematics and Applied Statistics
University of Wollongong, Australia

Abstract

Maximum likelihood (ML) estimation of spatial autocorrelation models is well-established for the case where each node in the graph is directly observed. When one or more nodes are not observed, the user has a variety of computational tools at her or his disposal ranging from the expectation-maximisation algorithm, which has become a standard for missing-data problems, marginal-likelihood estimation methods, and fully Bayesian approaches. In this article we give a comprehensive overview of likelihood-based computational frameworks for parameter estimation of the conditional-autoregressive model, and establish connections with several algorithms in the literature that are iterative and often computationally suboptimal. We show that a vanilla marginal ML approach, which we provide computational details for, is still generally orders of magnitude faster than the iterative approaches, even on large data sets and especially so when the number of unobserved units is relatively large.

Keywords: CAR model, EM algorithm, incomplete data, spatial statistics

1 Introduction

Spatial autoregressive models are popular regression models that are used for modelling data that are distributed on a discrete spatial domain D. There are two main types of such models, simultaneous autoregressive (SAR) models and conditional autoregressive (CAR) models, the former pioneered by Whittle (1954) and the latter by Besag (1974). Important SAR models are the spatial lag model, the spatial errors model and the spatial Durbin model (Lesage and Pace, 2009). In SAR models, spatial dependence in the response variable is accounted for by imposing distributional assumptions on elements of the domain D that are treated as neighbours. This neighbourhood structure is represented by a contiguity matrix $\mathbf{W}$ that has zeros on the diagonal and for which $W_{ij} = 1$ when i is a neighbour of j, for $i, j = 1, \ldots, n$, and n is the cardinality of D.

Several estimation methods have been developed for SAR models, the most popular of which are based on maximum likelihood (ML). A variety of computational approaches with the ML framework have also been considered. For example, for SAR models, Ord (1975) uses eigenvalues of the contiguity matrix $\mathbf{W}$ for computations inside an ML framework. However, as the number of operations associated with the calculation of eigenvalues is approximately $O(n^3)$, their approach quickly becomes prohibitive for large data sets. It is now commonplace to instead take advantage of the sparsity in $\mathbf{W}$, and to make extensive use of the sparse Cholesky factor of matrices constructed using $\mathbf{W}$ for estimation (Pace and Barry, 1997b). For CAR models, ML estimates are usually obtained through a two-step profile likelihood approach (e.g., Cressie and Wikle, 2011).

While computational methods for implementing ML estimation are well established for when every element in D is observed, these are less so for when only a subset of elements in $D^o \subset D$ is observed. For SAR models, Lesage and Pace (2004) considered an approximation to the expectation maximisation (EM) algorithm, while Suesse and Zammit-Mangion (2017) proposed various alternatives by approximating some of the terms inside the M-step of the algorithm. Kato (2013) considered the estimation of several spatial covariance models, including those in SAR models, using the EM algorithm and the quasi-likelihood method for estimation, while Goulard et al. (2017) used the EM algorithm for evaluating various in-sample and out-of-sample predictors for SAR models. Other estimation methods for SAR models with missing data have been extensively studied: These include the generalised method of moments and least squares approach (Wang and Lee, 2013), and integrated nested Laplace approximations (INLA; Bivand et al., 2014; Gómez-Rubio et al., 2017).

Current approaches for parameter-estimation with CAR models are generally iterative in nature. The most well-known of these is the EM algorithm, although other estimation methods can be found in the geostatistical literature. One influential method is that of Griffith et al. (1989) who considered an iterative method which replaces the missing data by their best linear unbiased predictor. On close inspection, their method, which is based on that of Martin (1984), is similar to the EM algorithm, an indirect maximiser of the marginal log-likelihood, but a true marginal ML method. Similar to the EM algorithm, their method is slow to converge and computationally inefficient.

This paper serves a dual purpose, first to clarify the connections between the algorithms of Martin (1984), Griffith et al. (1989) and the EM algorithm and, second, to establish the approach required to render marginal maximum-likelihood estimation methods in CAR models computationally efficient. Our results show that there is no benefit in considering iterative methods when doing ML estimation for CAR models, even when the data sets are large and especially so when the number of missing data elements is relatively large.

In Section 2 the CAR model is introduced. In Section 3 we outline the marginal ML method, the EM algorithm and the connections to the algorithms of Griffith et al. (1989) and Martin (1984). Section 4 establishes the computational details required to make the marginal ML method feasible. In Section 5 a simulation study is conducted to compare the EM algorithm, the marginal ML method and the methods proposed by Griffith et al. (1989) and Martin (1984). Section 6 illustrates the proposed method on a well known data set and the paper concludes with a brief discussion.

2 Conditional Autoregressive Models

Let $\mathbf{Y} = (Y_1, \ldots, Y_n)^\top$ be the n random vector of the response variable, $\mathbf{X}$ be the $n \times p$ design matrix containing the explanatory variables and $\mathbf{W}$ be the $n \times n$ contiguity matrix, where W_{ij} represents spatial adjacency of two units i and j; by convention $W_{ii} = 0, i = 1, \ldots, n$. Common methods to form $\mathbf{W}$ are through first order contiguity relations and nearest neighbours; see Ord (1975) or Pace and Barry (1997a) for more details. Often $\mathbf{W}$ is sparse in the sense that many of its elements are zero. The zeros often correspond to elements whose pairwise distance exceeds a fixed threshold.

Define $\mathbf{y}_{-i} = (y_1, \ldots, y_{i-1}, y_{i+1}, \ldots, y_n)^\top$ as the vector of responses excluding y_i. The CAR model is built from the following conditional relationships

$$Y_i|\mathbf{y}_{-i} \sim N\left(\sum_{i,j} C_{ij} y_j, \Delta_{ii}\right), \quad i = 1, \ldots, n, \tag{1}$$

where C_{ij} are the elements of the spatial dependence matrix $\mathbf{C}$ with $C_{ii} = 0$, and the diagonal matrix $\mathbf{\Delta}$ has positive elements Δ_{ii}, which may depend on $\mathbf{C}$. The conditional mean of Y_i depends on the values of all the neighbours of the ith node, that is, on all y_j with $C_{ij} > 0$.

Under certain conditions, the conditional specification of the CAR model in Equation (1) also fully specifies a valid joint distribution of $\mathbf{Y}$. Specifically, Besag (1974) showed that the random vector $\mathbf{Y}$ has zero mean vector and (co)variance matrix $\text{Cov}(\mathbf{Y}) \equiv \mathbf{\Sigma} = (\mathbf{I} - \mathbf{C})^{-1}\mathbf{\Delta}$, provided $(\mathbf{I} - \mathbf{C})^{-1}\mathbf{\Delta}$ is positive definite. The matrix $\mathbf{\Sigma}$ must also be symmetric, that is,

$$\frac{C_{ij}}{\Delta_{ii}} = \frac{C_{ji}}{\Delta_{jj}}. \tag{2}$$

If $\mathbf{\Delta}$ is positive definite then it is sufficient for $\mathbf{I} - \mathbf{C}$ to have positive eigenvalues for $\mathbf{\Sigma}$ to be positive definite. The following are sufficient conditions to establish a valid CAR model (Hoef et al., 2017):

(C1) $\mathbf{I} - \mathbf{C}$ has positive eigenvalues,

(C2) $\mathbf{\Delta}$ has positive diagonal entries, that is, $\Delta_{ii} > 0, i = 1, \ldots, n$,

(C3) $C_{ii} = 0, i = 1, \ldots, n$, and

(C4) Equation (2) holds.

Often the practitioner aims to include covariates in the model of the mean. In this setting, the general CAR model has the form

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{e}, \tag{3}$$

where $\mathbf{X}$ is a matrix of covariates with ones along the first column, $\boldsymbol{\beta}$ is a p-dimensional vector of regression coefficients, and $\mathbf{e}$ is a zero-mean CAR model. It then follows that $\mathbf{Y} \sim N(\boldsymbol{\mu} \equiv \mathbf{X}\boldsymbol{\beta}, \mathbf{\Sigma} \equiv (\mathbf{I} - \mathbf{C})^{-1}\mathbf{\Delta})$. Usually, the contiguity matrix $\mathbf{W}$ is itself used to model $\mathbf{C}$; the simplest model being $\mathbf{C} = \rho\mathbf{W}$ where ρ is an unknown parameter that needs to be estimated. Usually $\mathbf{\Delta}$ is also modelled as $\sigma^2\tilde{\mathbf{\Delta}}$ with known $\tilde{\mathbf{\Delta}}$, that is, $\mathbf{\Delta}$ is assumed known up to a constant σ^2, which needs to be estimated. Then, $\mathbf{\Sigma} = \sigma^2(\mathbf{I} - \rho\mathbf{W})^{-1}\tilde{\mathbf{\Delta}}$. Let $\lambda_{(1)} \leq \cdots \leq \lambda_{(n)}$ be the ordered n eigenvalues of the matrix $\mathbf{W}$. Then $\frac{1}{\lambda_{(1)}} < \rho < \frac{1}{\lambda_{(n)}}$ ensures that $\mathbf{I} - \rho\mathbf{W}$ has positive eigenvalues and that condition C1 is met.

In this paper we aim to make inference at n units when only $n_s < n$ units are observed. We assume that data are missing at random for ML estimation to be consistent.

3 Maximising the Marginal Likelihood

3.1 Marginal Maximum Likelihood

Let $\omega \equiv \sigma^2$ and $\boldsymbol{\Sigma} \equiv \omega \mathbf{V}$, that is, let $\mathbf{V} = (\mathbf{I} - \rho \mathbf{W})^{-1} \tilde{\boldsymbol{\Delta}}$. Instead of maximising the marginal log-likelihood indirectly via the EM algorithm, (see Suesse and Zammit-Mangion, 2017; Lesage and Pace, 2004), we consider marginal ML (see Suesse, 2018, for the case where marginal ML estimation was considered for SAR models). We will follow closely the notation used in Suesse (2018), using $\mathbf{M}$ to denote the precision matrix, that is, $\mathbf{M} \equiv \mathbf{V}^{-1} = \tilde{\boldsymbol{\Delta}}^{-1}(\mathbf{I} - \rho \mathbf{W})$.

Let s be the set of units that are observed and u be the set of units that are unobserved. Vectors (e.g., $\boldsymbol{\mu}$) and matrices (e.g., $\mathbf{M}$) are partitioned as follows:

$$\boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_s \\ \boldsymbol{\mu}_u \end{pmatrix}, \qquad \mathbf{M} = \begin{pmatrix} \mathbf{M}_{ss} & \mathbf{M}_{su} \\ \mathbf{M}_{us} & \mathbf{M}_{uu} \end{pmatrix}. \tag{4}$$

The random vector $\mathbf{y}$ is multivariate normal, and consequently $\mathbf{y}_s$ is also multivariate normal with mean $\boldsymbol{\mu}_s$ and variance $\omega \mathbf{V}_{ss}$. Hence, the marginal log-likelihood of $\mathbf{y}_s$ with parameter vector $\boldsymbol{\theta} = (\boldsymbol{\beta}^\top, \omega, \rho)^\top$ is given by

$$\log f(\mathbf{y}_s; \boldsymbol{\theta}) = -\frac{n_s}{2} \log(2\pi) - \frac{n_s}{2} \log \omega + \frac{1}{2} \log |\mathbf{V}_{ss}^{-1}| - \frac{1}{2\omega} \mathbf{r}_s^\top \mathbf{V}_{ss}^{-1} \mathbf{r}_s, \tag{5}$$

where $\mathbf{r}_s = \mathbf{y}_s - \boldsymbol{\mu}_s$.

Maximising (5) with respect to $\boldsymbol{\beta}$ and ω for fixed ρ gives the ML estimates

$$\hat{\boldsymbol{\beta}}(\rho) = (\mathbf{X}_s^\top \mathbf{V}_{ss}^{-1} \mathbf{X}_s)^{-1} \mathbf{X}_s^\top \mathbf{V}_{ss}^{-1} \mathbf{y}_s, \text{ and } \hat{\omega}(\rho) = \frac{1}{n_s} \hat{\mathbf{r}}_s^\top \mathbf{V}_{ss}^{-1} \hat{\mathbf{r}}_s, \tag{6}$$

where $\hat{\mathbf{r}}_s \equiv \mathbf{y}_s - \mathbf{X}_s \hat{\boldsymbol{\beta}}(\rho)$. Plugging these estimates back into (5) provides the concentrated log-likelihood that only depends on the parameter ρ,

$$\begin{aligned} \log f(\mathbf{y}_s; \rho) &= -\frac{n_s}{2} \log(2\pi) - \frac{n_s}{2} \log \frac{\hat{\mathbf{r}}_s^\top \mathbf{V}_{ss}^{-1} \hat{\mathbf{r}}_s}{n_s} + \frac{1}{2} \log |\mathbf{V}_{ss}^{-1}| - \frac{n_s}{2} \\ &= -\frac{n_s}{2} \left(\log \left(\frac{2\pi}{n_s} \right) + 1 \right) - \frac{n_s}{2} \log \hat{\mathbf{r}}_s^\top \mathbf{V}_{ss}^{-1} \hat{\mathbf{r}}_s + \frac{1}{2} \log |\mathbf{V}_{ss}^{-1}|. \end{aligned} \tag{7}$$

Martin (1984, p. 1278) suggested minimising the expression

$$|\mathbf{V}_{ss}|^{\frac{1}{n_s}} \left(\hat{\mathbf{r}}_s^\top \mathbf{V}_{ss}^{-1} \hat{\mathbf{r}}_s \right),$$

which is equivalent to maximising the concentrated log likelihood (7); see Appendix A for details. Estimation of ρ thus reduces to a one-dimensional optimisation problem, for which extensive software and algorithms are available. (In our experiments we used the `R` routine `optimize()`; see Brent, 1973). As long as the terms $\hat{\mathbf{r}}_s^\top \mathbf{V}_{ss}^{-1} \hat{\mathbf{r}}_s$ and $\log |\mathbf{V}_{ss}^{-1}|$ can be calculated quickly, the underlying optimisation problem is indeed straightforward.

Consider now the formula of the inverse of a partitioned matrix (see, for example, Harville, 1997),

$$\mathbf{M}^{-1} = \begin{pmatrix} (\mathbf{M}_{ss} - \mathbf{M}_{su} \mathbf{M}_{uu}^{-1} \mathbf{M}_{us})^{-1} & -\mathbf{M}_{ss}^{-1} \mathbf{M}_{su} (\mathbf{M}_{uu} - \mathbf{M}_{us} \mathbf{M}_{ss}^{-1} \mathbf{M}_{su})^{-1} \\ -\mathbf{M}_{uu}^{-1} \mathbf{M}_{us} (\mathbf{M}_{ss} - \mathbf{M}_{su} \mathbf{M}_{uu}^{-1} \mathbf{M}_{us})^{-1} & (\mathbf{M}_{uu} - \mathbf{M}_{us} \mathbf{M}_{ss}^{-1} \mathbf{M}_{su})^{-1} \end{pmatrix}.$$

Then, since $\mathbf{V} = \mathbf{M}^{-1}$, the formulae for the mean $\boldsymbol{\mu}_s$ and variance $\mathbf{V}_{ss}$ are

$$\begin{aligned} \boldsymbol{\mu}_s &= \mathbf{X}_s \boldsymbol{\beta}, \\ \mathbf{V}_{ss} &= (\mathbf{M}_{ss} - \mathbf{M}_{su} \mathbf{M}_{uu}^{-1} \mathbf{M}_{us})^{-1}, \end{aligned} \tag{8}$$

and therefore $\mathbf{V}_{ss}^{-1}$ takes the relatively simple form

$$\mathbf{V}_{ss}^{-1} = \mathbf{M}_{ss} - \mathbf{M}_{su} \mathbf{M}_{uu}^{-1} \mathbf{M}_{us}. \tag{9}$$

Notice that $\mathbf{V}_{ss}^{-1} \neq \mathbf{M}_{ss}$ unless $\mathbf{M}_{su} = \mathbf{W}_{su} = \mathbf{0}$. Marginal ML estimation follows by directly maximising (7) as a function of ρ. We denote the marginal ML method as MML.

3.2 EM Algorithm

Let $\boldsymbol{\theta} = (\boldsymbol{\beta}^\top, \omega, \rho)^\top$ be the vector of parameters of the CAR model and also let $\boldsymbol{\theta}'$ be the a current estimate of $\boldsymbol{\theta}$. The EM algorithm alternates between an E-step and an M-step. In the E-step, the following expectation is calculated,

$$E_{u|s}(\mathbf{y}_u; \boldsymbol{\theta}') \equiv \boldsymbol{\mu}_{u|s} = \boldsymbol{\mu}_u + \mathbf{V}_{us} \mathbf{V}_{ss}^{-1} (\mathbf{y}_s - \boldsymbol{\mu}_s), \tag{10}$$

which is computationally more conveniently expressed in terms of $\mathbf{M}$ as

$$E_{u|s}(\mathbf{y}_u;\boldsymbol{\theta}') = \boldsymbol{\mu}_u - \mathbf{M}_{uu}^{-1}\mathbf{M}_{us}(\mathbf{y}_s - \boldsymbol{\mu}_s). \tag{11}$$

In the M-step, the Q-function $Q(\boldsymbol{\theta}|\boldsymbol{\theta}') \equiv E_{u|s}\log f(\mathbf{y};\boldsymbol{\theta})$ is maximised with respect to $\boldsymbol{\theta}$ to yield a new estimate of $\boldsymbol{\theta}$, where

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}') = -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log\omega + \frac{1}{2}\log|\mathbf{M}(\rho)| - \frac{\mathbf{r}_{u|s}^\top\mathbf{M}(\rho)\mathbf{r}_{u|s} + \omega'\mathrm{tr}\left\{\mathbf{M}_{uu}(\rho')^{-1}\mathbf{M}_{uu}(\rho)\right\}}{2\omega},$$

$\mathbf{r}_{u|s} = \mathbf{r}_{u|s}(\boldsymbol{\theta}|\boldsymbol{\theta}') \equiv E_{u|s}(\mathbf{y};\boldsymbol{\theta}') - \boldsymbol{\mu}(\boldsymbol{\theta})$, and $E_{u|s}(\mathbf{y};\boldsymbol{\theta}') = (\mathbf{y}_s^\top, \boldsymbol{\mu}_{u|s}^\top)^\top$.

The M-step can be expressed solely in terms of ρ by calculating the maximisers for $\boldsymbol{\beta}$ and ω analytically, given by

$$\hat{\boldsymbol{\beta}}(\rho|\boldsymbol{\theta}') = \left(\mathbf{X}^\top\mathbf{M}(\rho)\mathbf{X}\right)^{-1}\mathbf{X}^\top\mathbf{M}(\rho)E_{u|s}(\mathbf{y};\boldsymbol{\theta}'), \tag{12}$$

$$\hat{\omega}(\rho|\boldsymbol{\theta}') = \frac{1}{n}\mathbf{r}_{u|s}^\top\mathbf{M}(\rho)\mathbf{r}_{u|s}, \tag{13}$$

and then plugging these into the Q-function. This leads to the so-called *concentrated* Q-function

$$Q(\rho|\boldsymbol{\theta}') = -\frac{n}{2}(\log(2\pi)+1) - \frac{n}{2}\log\hat{\omega}(\rho|\boldsymbol{\theta}') + \frac{1}{2}\log|\mathbf{M}(\rho)| - \frac{\omega'\mathrm{tr}\left\{\mathbf{M}_{uu}(\rho')^{-1}\mathbf{M}_{uu}(\rho)\right\}}{2\hat{\omega}(\rho|\boldsymbol{\theta}')}, \tag{14}$$

which is only a function of ρ.

3.3 Other iterative methods

Martin (1984, p. 1280) showed that

$$\mathbf{r}_s^\top\mathbf{V}_{ss}^{-1}\mathbf{r}_s \equiv \mathbf{r}_{u|s}^\top\mathbf{V}^{-1}\mathbf{r}_{u|s}. \tag{15}$$

Based on this equivalence, the author suggested to replace $\mathbf{r}_s^\top\mathbf{V}_{ss}^{-1}\mathbf{r}_s$ in Equation (5) with $\mathbf{r}_{u|s}^\top\mathbf{V}^{-1}\mathbf{r}_{u|s}$, which is a computationally efficient way to avoid dealing with $\mathbf{V}_{ss}^{-1}$ and, rather, dealing with the computationally simpler precision matrix $\mathbf{V}^{-1} = \mathbf{M}$. After this substitution, expression (5) becomes

$$\log f(\mathbf{y}_s;\boldsymbol{\theta}) = -\frac{n_s}{2}\log(2\pi) - \frac{n_s}{2}\log\omega + \frac{1}{2}\log|\mathbf{V}_{ss}^{-1}| - \frac{1}{2\omega}\mathbf{r}_{u|s}^\top\mathbf{M}\mathbf{r}_{u|s}. \tag{16}$$

Maximising this expression with respect to $\boldsymbol{\beta}$ and ω gives

$$\hat{\boldsymbol{\beta}}(\rho) = \left(\mathbf{X}^\top\mathbf{M}(\rho)\mathbf{X}\right)^{-1}\mathbf{X}^\top\mathbf{M}(\rho)E_{u|s}(\mathbf{y};\rho), \quad \text{and} \tag{17}$$

$$\hat{\omega}(\rho) = \frac{1}{n_s}\mathbf{r}_{u|s}(\rho)^\top\mathbf{M}(\rho)\mathbf{r}_{u|s}(\rho). \tag{18}$$

Plugging these back into (16) yields

$$\log f(\mathbf{y}_s;\rho) = -\frac{n_s}{2}\left(\log(2\pi)+1\right) - \frac{n_s}{2}\log\hat{\omega}(\rho) + \frac{1}{2}\log|\mathbf{V}_{ss}^{-1}(\rho)|. \tag{19}$$

Instead of maximising (19) as a function of ρ, Martin (1984, p. 1281) suggested an iterative algorithm. Given the ρ estimate from the previous iteration, denoted by ρ', we can obtain the estimates of the regression coefficients through

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{X}^\top\mathbf{M}(\rho')\mathbf{X}\right)^{-1}\mathbf{X}^\top\mathbf{M}(\rho')E_{u|s}(\mathbf{y};\rho'). \tag{20}$$

Martin (1984) then proposed that $\hat{\rho}$ be found by minimising

$$|\mathbf{V}(\rho)|^{\frac{1}{n_s}} \times \mathbf{r}_{u|s}^\top\mathbf{M}(\rho)\mathbf{r}_{u|s}, \tag{21}$$

where $\mathbf{r}_{u|s} = \mathbf{r}_{u|s}(\rho', \hat{\boldsymbol{\beta}}) = E_{u|s}(\mathbf{y};\rho') - \mathbf{X}\hat{\boldsymbol{\beta}}$. When ω is unknown then its estimate is obtained through

$$\hat{\omega} = \frac{1}{n_s}\mathbf{r}_{u|s}(\rho', \hat{\boldsymbol{\beta}})^\top\mathbf{M}(\hat{\rho})\mathbf{r}_{u|s}(\rho', \hat{\boldsymbol{\beta}}). \tag{22}$$

The algorithm proceeds by iteratively setting $\rho' = \hat{\rho}$ and computing Equations (20)-(22) until convergence is reached.

It is likely that Martin (1984) made a small mistake in Equation (21) by using $|\mathbf{V}(\rho)|^{\frac{1}{n_s}}$ instead of $|\mathbf{V}_{ss}(\rho)|^{\frac{1}{n_s}}$. Griffith et al. (1989, p. 1517) used a correct version, and proposed the ensuing algorihm for CAR models (and also one for the spatial

errors model) that is initialised with $\rho' = 0$. Griffith et al. (1989) do not compute (22) since Equations (20) and (21) do not depend on $\hat{\omega}$. Instead, $\hat{\omega}$ is obtained after the algorithm has converged and produces an estimate of ρ, which then uniquely determines $\hat{\omega}$ and $\hat{\boldsymbol{\beta}}$.

In the following we refer to the algorithm by Griffith et al. (1989) as GBH89 and to the algorithm by Martin (1984) as Martin84. The drawback of these algorithms is that both require (usually) several iterations until convergence is achieved. Further, this indirect maximisation is unnecessary since (19) is a function of ρ and can be maximised directly. The GBH89 and Martin84 algorithms in fact could be classified as coordinate-ascent optimisation methods, which are known to be slower to converge to a local maximum than standard ascent-based methods.

In Section 4 we present details on how to efficiently compute the terms needed in all the algorithms. These include the terms for Martin84 and GBH89 since both Martin (1984) and Griffith et al. (1989) did not take advantage of sparse matrix linear algebraic operations, implementations for which are nowadays widely available. In Section 5 we then conduct a simulation study comparing the computational efficiency of the discussed approaches.

4 Computational Aspects

It is unclear why iterative methods for ML estimation with partially-observed CAR models have garnered so much interest in the spatial econometrics and geostatistics community. One reason could be that considerable care in the implementation must be taken when estimating using MML, to ensure computational feasibility. In this section we establish the computational details required to implement the MML method efficiently.

Let $\widetilde{\mathbf{Y}} \equiv \tilde{\boldsymbol{\Delta}}^{-1/2}\mathbf{Y}$ and assume that the conditional variance is known up to a constant, that is, $\mathrm{Var}(Y_i|\mathbf{y}_{-i}) = \sigma^2\tilde{\Delta}_{ii}, i = 1,\dots,n$. Then the mean and the covariance matrix of $\widetilde{\mathbf{Y}}$ are $\widetilde{\boldsymbol{\mu}} = \tilde{\boldsymbol{\Delta}}^{-1/2}\mathbf{X}\boldsymbol{\beta} = \widetilde{\mathbf{X}}\boldsymbol{\beta}$ and

$$\begin{aligned}\widetilde{\boldsymbol{\Sigma}} &= \sigma^2\tilde{\boldsymbol{\Delta}}^{-1/2}(\mathbf{I}-\rho\mathbf{W})^{-1}\tilde{\boldsymbol{\Delta}}\tilde{\boldsymbol{\Delta}}^{-1/2}\\ &= \sigma^2\left[\tilde{\boldsymbol{\Delta}}^{-1/2}(\mathbf{I}-\rho\mathbf{W})\tilde{\boldsymbol{\Delta}}^{1/2}\right]^{-1}\\ &= \sigma^2(\mathbf{I}-\rho\tilde{\boldsymbol{\Delta}}^{-1/2}\mathbf{W}\tilde{\boldsymbol{\Delta}}^{1/2})^{-1}\\ &= \sigma^2(\mathbf{I}-\rho\widetilde{\mathbf{W}})^{-1}.\end{aligned}$$

Hence, by transforming the process and using $\widetilde{\mathbf{Y}}$, $\widetilde{\mathbf{X}} \equiv \tilde{\boldsymbol{\Delta}}^{-1/2}\mathbf{X}$ and $\widetilde{\mathbf{W}} \equiv \tilde{\boldsymbol{\Delta}}^{-1/2}\mathbf{W}\tilde{\boldsymbol{\Delta}}^{1/2}$ we obtain a general CAR model with $\boldsymbol{\Delta} = \sigma^2\mathbf{I}$. Therefore, without loss of generality, in this section we assume that $\boldsymbol{\Delta} = \sigma^2\mathbf{I}$. For this model, $\mathbf{M} = \mathbf{I} - \rho\widetilde{\mathbf{W}}$.

The calculation of (9) involves the calculation of $\mathbf{M}_{uu}^{-1}$ which should not be computed explicitly (especially) for large n_u. Let

$$\begin{aligned}\mathbf{a} &= \rho\mathbf{L}_{uu}^{-1}\mathbf{W}_{us}\mathbf{X}_s,\\ \mathbf{b} &= \rho\mathbf{L}_{uu}^{-1}\mathbf{W}_{us}\mathbf{y}_s,\end{aligned} \tag{23}$$

where $\mathbf{L}_{uu}$ is the lower Cholesky factor of $\mathbf{M}_{uu} = \mathbf{I}_u - \rho\mathbf{W}_{uu}$. Since $\mathbf{W}_{us}\mathbf{X}_s$ and $\mathbf{W}_{us}\mathbf{y}_s$ are of dimensions $n_u \times p$ and $n_u \times 1$, respectively, the calculation of $\mathbf{a}$ and $\mathbf{b}$ only requires $(n_u \times (p+1))$ sparse forward solves that are generally fast to compute, even for large n_u.

The MML computations for partially-observed CAR models proceed in two steps. In the first step ρ is fixed and $\hat{\boldsymbol{\beta}}$ is calculated from $\mathbf{X}_s^\top\mathbf{V}_{ss}^{-1}\mathbf{X}_s$ and $\mathbf{X}_s^\top\mathbf{V}_{ss}^{-1}\mathbf{y}_s$. In the second step $\hat{\omega}$ is calculated. This second step requires computaton of $\hat{\mathbf{r}}_s^\top\mathbf{V}_{ss}^{-1}\hat{\mathbf{r}}_s$, which can be expressed as

$$\begin{aligned}\hat{\mathbf{r}}_s^\top\mathbf{V}_{ss}^{-1}\hat{\mathbf{r}}_s &= \mathbf{y}_s^\top\mathbf{V}_{ss}^{-1}\mathbf{y}_s - 2\hat{\boldsymbol{\beta}}^\top\mathbf{X}_s^\top\mathbf{V}_{ss}^{-1}\mathbf{y}_s + \hat{\boldsymbol{\beta}}^\top(\mathbf{X}_s^\top\mathbf{V}_{ss}^{-1}\mathbf{X}_s)\hat{\boldsymbol{\beta}}\\ &= \mathbf{y}_s^\top\mathbf{V}_{ss}^{-1}\mathbf{y}_s - \hat{\boldsymbol{\beta}}^\top\mathbf{X}_s^\top\mathbf{V}_{ss}^{-1}\mathbf{y}_s.\end{aligned}$$

The three terms $\mathbf{X}_s^\top\mathbf{V}_{ss}^{-1}\mathbf{X}_s$, $\mathbf{X}_s^\top\mathbf{V}_{ss}^{-1}\mathbf{y}_s$ and $\mathbf{y}_s^\top\mathbf{V}_{ss}^{-1}\mathbf{y}_s$ that are required can be expressed as

$$\mathbf{X}_s^\top\mathbf{V}_{ss}^{-1}\mathbf{X}_s = \mathbf{X}_s^\top\mathbf{M}_{ss}\mathbf{X}_s - \mathbf{a}^\top\mathbf{a}, \tag{24}$$

$$\mathbf{X}_s^\top\mathbf{V}_{ss}^{-1}\mathbf{y}_s = \mathbf{X}_s^\top\mathbf{M}_{ss}\mathbf{y}_s - \mathbf{a}^\top\mathbf{b}, \quad \text{and} \tag{25}$$

$$\mathbf{y}_s^\top\mathbf{V}_{ss}^{-1}\mathbf{y}_s = \mathbf{y}_s^\top\mathbf{M}_{ss}\mathbf{y}_s - \mathbf{b}^\top\mathbf{b}. \tag{26}$$

The terms $\mathbf{X}_s^\top\mathbf{M}_{ss}\mathbf{X}_s$, $\mathbf{X}_s^\top\mathbf{M}_{ss}\mathbf{y}_s$ and $\mathbf{y}_s^\top\mathbf{M}_{ss}\mathbf{y}_s$ can be simplified further. In particular,

$$\mathbf{y}_s^\top\mathbf{M}_{ss}\mathbf{y}_s = \mathbf{y}_s^\top\mathbf{y}_s - \rho\mathbf{y}_s^\top\mathbf{W}_{ss}\mathbf{y}_s,$$

which can be calculated efficiently for a given value of ρ, as the two terms $\mathbf{y}_s^\top\mathbf{y}_s$ and $\mathbf{y}_s\mathbf{W}_{ss}\mathbf{y}_s$ do not depend on ρ and thus only need to be computed once.

Table 1: Computational results: Average negative log-likelihood ($-l$) on convergence, average number of iterations and average computational time required (in seconds) for the considered methods across 10,000 simulations.

Average negative log-likelihood ($-l$) on convergence

n_s	20	50	100	200	300	400	500
MML	27.172	70.720	143.364	287.849	431.459	574.451	716.684
EM	27.216	70.721	143.364	287.849	431.459	574.451	716.684
Martin84	30.056	76.299	153.117	295.350	433.924	574.886	716.685
GBH89	27.192	70.721	143.364	287.849	431.459	574.451	716.684

Average number of iterations

n_s	20	50	100	200	300	400	500
MML	N/A	N/A	N/A	N/A	N/A	N/A	N/A
EM	897.8	517.3	233.4	68.7	31.0	15.5	5.5
Martin84	754.9	262.8	137.9	43.7	26.5	14.4	5.9
GBH89	450.0	192.1	103.9	45.9	25.3	13.9	6.0

Average computational time required (in seconds)

n_s	20	50	100	200	300	400	500
MML	0.39	0.33	0.31	0.30	0.30	0.30	0.32
EM	352.71	172.94	60.74	11.73	5.17	2.70	0.93
Martin84	116.88	39.64	18.89	4.37	2.39	1.29	0.49
GBH89	57.43	21.35	10.16	4.13	2.28	1.29	0.53
INLA	4.55	5.25	5.82	6.82	7.00	7.04	6.87

Finally, the term $|\mathbf{V}_{ss}|$ or $|\mathbf{V}_{ss}^{-1}|$ has to be calculated. While so far the calculation of $\mathbf{M}_{uu}^{-1}$ has been avoided, (9) requires the calculation of $\mathbf{M}_{su}\mathbf{M}_{uu}^{-1}\mathbf{M}_{us}$ which can be costly if n_s or n_u are large. To circumvent this problem we exploit a standard linear algebra result (Harville, 1997), which establishes that the determinant of the matrix $\mathbf{M}$ can be expressed as the determinant of its block matrix components

$$|\mathbf{M}| = |\mathbf{M}_{uu}| \times |\mathbf{M}_{ss} - \mathbf{M}_{su}\mathbf{M}_{uu}^{-1}\mathbf{M}_{us}|.$$

Therefore, we have that

$$\log|\mathbf{V}_{ss}^{-1}| = \log|\mathbf{M}| - \log|\mathbf{M}_{uu}| = \log|\mathbf{I} - \rho\mathbf{W}| - \log|\mathbf{I}_u - \rho\mathbf{W}_{uu}|. \tag{27}$$

This result was also mentioned by Martin (1984, p. 1279) and the first term in (27) is used in the `R` package `spdep` (Bivand, 2018) for fitting CAR and SAR models. Instead of computing $\log|\mathbf{M}|$ and $\log|\mathbf{M}_{uu}|$ for each value of ρ, as is done for standard SAR models (Suesse, 2018), we can further improve computational efficiency by re-expressing $|\mathbf{I} - \rho\mathbf{W}|$ (and similarly $|\mathbf{I}_u - \rho\mathbf{W}_{uu}|$) as

$$|\mathbf{I} - \rho\mathbf{W}| = \begin{cases} (\rho)^n \left|\frac{1}{\rho}\mathbf{I} + (-\mathbf{W})\right|; & \rho > 0, \\ (-\rho)^n \left|\left(-\frac{1}{\rho}\right)\mathbf{I} + \mathbf{W}\right|; & \rho < 0. \end{cases}$$

Using this representation, one can leverage efficient updating algorithms that obviate the need to compute Cholesky factorisations for each value of ρ. For example, one could make use of the updating algorithm available in the `ldetL2up()` function in the `R` package `Matrix` (Bates and Maechler, 2018; Chen et al., 2008). Specifically, suppose that the Cholesky factorisation of $c\mathbf{I} + \mathbf{W}$ is available, then one can update the Cholesky factorisation of $c\mathbf{I} + \mathbf{W}$ to obtain the Cholesky factorisation of $d\mathbf{I} + \mathbf{W}$ for any value of d. In our case $d = \frac{1}{\rho}$ when $\rho > 0$ and $d = -\frac{1}{\rho}$ when $\rho < 0$. This updating process is usually more computationally efficient than one using factorisation.

Note that although the Cholesky factor of $\mathbf{M}$ need only be computed once using this procedure, that of $\mathbf{M}_{uu}$ needs to be computed for each value of ρ to compute (23) and (27). This does mean that the MML algorithm will slow as n_u increases, although iterative methods also tend to converge very slowly for large n_u. This burden can be alleviated to a large extent by instead computing the factor of a fill-in reducing permutation of $\mathbf{M}_{uu}$; see Rue and Held (2005, Ch. 2) for details.

Closed-form expressions are available for the expected information of the parameters associated with the CAR model. We provide these in Appendix B.

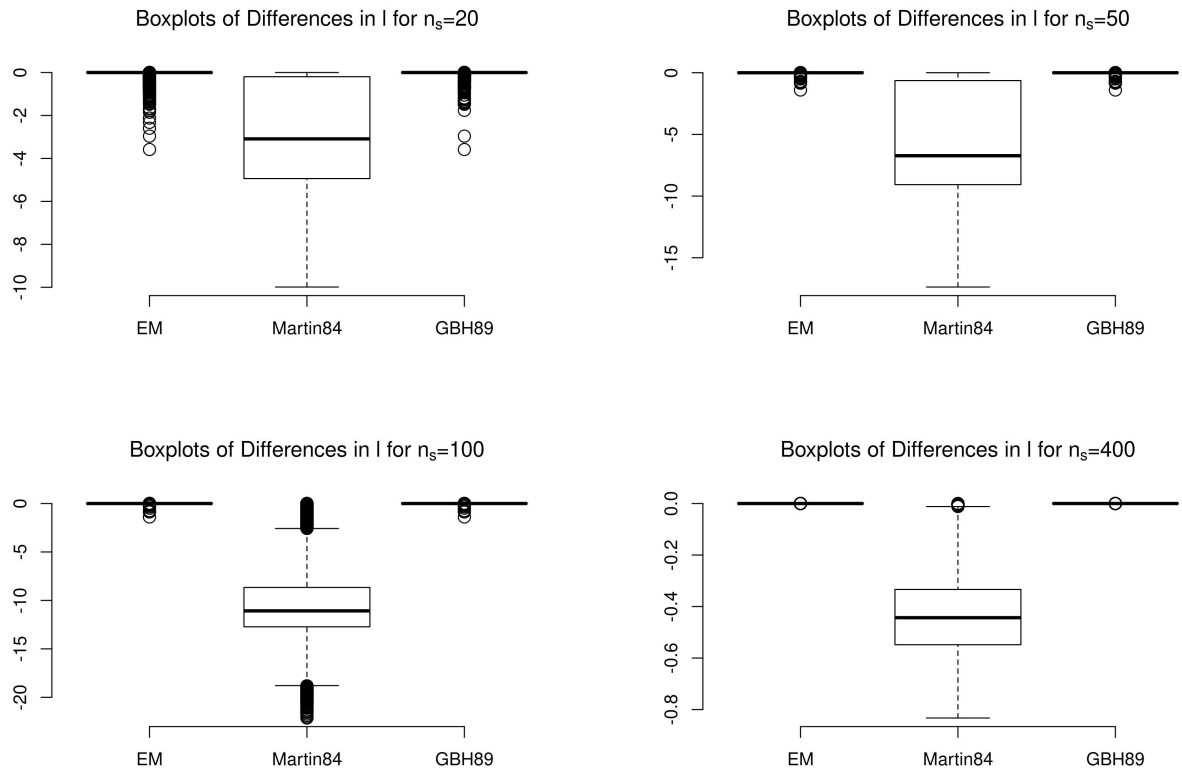

Figure 1: Boxplots of differences between the log-likelihood (l) of an algorithm (EM, Martin84, or GBH89) and that obtained through MML estimation on convergence for $n_s \in \{20, 50, 100, 400\}$ (see text for details).

5 Simulation Study

In this section we empirically compare the discussed methods in terms of computational time and convergence, namely MML estimation, the EM algorithm, Martin84, and GBH89. We use the contiguity matrix $\mathbf{W}$ from the Corrected Boston Housing Data, collected by Harrison and Rubinfeld (1978) and available in the `R` package `spdep` (Bivand, 2018). For this study we set $\mathbf{\Delta} = \sigma^2\mathbf{I}$ and $\mathbf{C} = \rho\overline{\mathbf{W}}$, where $\overline{\mathbf{W}}$ for the $n = 506$ units was obtained through $\overline{\mathbf{W}} = \mathbf{D}^{-1/2}\mathbf{W}\mathbf{D}^{-1/2}$, where $\mathbf{D}$ is a diagonal matrix with elements $D_{ii} = \sum_{j=1}^{n} W_{ij}$ (the number of neighbours of unit i). Under this normalisation, the largest eigenvalue $\lambda_{(n)} = 1$, and hence the upper bound for ρ is one.

The data were generated using a general CAR model with intercept $\beta_0 = 1$ and slope $\beta_1 = 2$, the spatial dependence parameter was set to $\rho = 0.5$ and the variance to $\omega \equiv \sigma^2 = 1$. The elements of $\mathbf{x}_1$ in $\mathbf{X} \equiv (\mathbf{1}, \mathbf{x}_1)$, were sampled from a standard normal distribution. To implement MML we used the `R` routine `optimise()` to maximise the log-likelihood (5) with a tolerance of 10^{-8} (i.e., the optimisation was set to terminate when $\sum_j |\theta_j - \theta'_j| < 10^{-8}$). The function `optimise()` was also used for the M-step of the EM algorithm, and for Equation (21) in both Martin84 and GBH89. For these iterative algorithms, convergence was deemed to be reached when $\sum_j |\theta_j - \theta'_j| < 10^{-4}$. This higher tolerance was required for the iterative algorithms to reach convergence within reasonable time frames. All experiments were run on a multi-core machine containing Intel Xeon E5-2620 2.10GHz processors.

Table 1 shows the values of the empirical mean of the value of the log-likelihood, the number of iterations of the iterative methods and the computation times in seconds, for various values of n_s (n_s = 20, 50, 100, 200, 300, 400, 500) when fitting CAR models to 10,000 realisations of $\mathbf{Y}$. The results show that MML is orders of magnitude faster than the iterative methods when n_s is small, and still considerably faster when $n_s \approx n$. Among the iterative methods the GBH89 algorithm is fastest, while the Martin84 algorithm does not converge to the correct ML value in most cases, as expected. Due to the popularity of INLA in fitting these types of models (e.g., Bivand et al., 2015), we also show timings for INLA when using the `generic1` model with a Gaussian likelihood with fixed, large, measurement-error precision (note, however, that INLA is based on approximate Bayesian computations and uses multiple cores, and that the computational complexity for a latent Gaussian model of this type is largely determined by n and not n_s).

To empirically compare the rate of convergence of the various algorithms we show box plots of the differences between the log-likelihood of the iterative methods and that from MML estimation, on convergence, in Figure 1. A value of zero indicates that an algorithm achieved the same log-liklihood as MML estimation, while a negative value indicates that an

algorithm achieved a lower log-likelihood. From the boxplots it can be seen that for large n_s (for example $n_s = 400$) the EM algorithm, GBH89 and MML all converge to the same log-likelihood value. For small n_s, the EM algorithm and GBH89 do not, in general, outperform MML, and often perform worse, especially for small n_s.

6 Real-data Example

The Lucas County (Ohio, USA) housing data set consists of $n = 25,357$ observations of single-family homes sold in the period 1993–1998. The data set is part of the `R` package `spdep` (Bivand, 2018) and is comprehensively described in the Spatial Econometrics toolbox for Matlab; see `http://www.spatial-econometrics.com/html/jplv7.zip`. The data have been used by Bivand (2010) to compare several software packages for fitting spatial regression models. Bivand (2010) used as response variable the log house price ($\log(price)$), and as explanatory variables powers of age (age, age^2 and age^3), log lot size in square feet ($\log(lotsize)$), the number of rooms ($rooms$), the log of the total living area in square feet (LTA), the number of $beds$, and an indicator for each of the years 1993–1998 ($syear$). Here we use the same binary sparse contiguity matrix $\mathbf{W}$ as used by Bivand (2010), but again transformed to $\overline{\mathbf{W}}$ as in Section 5.

To obtain a data set with missing data, a systematic sample was produced of size $n_s = 1,015$ by selecting every 25th data point: $\mathbf{y}_s = (y_1, y_{26}, y_{51}, \ldots, y_{25351})^\top$. Fitting a CAR model to the full data set (i.e., with no unobserved data) using the package `spdep` took less than 1 second while, using the convergence criteria detailed in Section 5, MML estimation took approximately 3.5 seconds with missing data. The iterative methods proved problematic with such a large data set, despite the use of the sparse matrix algebra operations detailed in Section 4. GBH89 gave the same results as MML estimation, but convergence required more than 4 hours. The EM algorithm was stopped after 10,000 iterations, at which point it had still not converged, while Martin84 did not give converge to sensible estimates, as expected. For completeness, the results of the fully observed data are given in Table 2. Note how MML estimation returned reasonable estimates for all parameters (when compared to those using all the data) despite using only 4% of the data (but the same underlying neighbourhood graph of size n).

Table 2: CAR model estimates and standard errors (in brackets) for a sample of size $n_s = 1,015$ using MML estimation, the EM algorithm, Martin84 and GBH89. The CAR model estimates when the full data set is fitted using `spdep` are also shown. Note that standard errors for ω are not available from `spdep`.

	MML	EM	Martin84	GBH89	Full data
Intercept	4.171 (0.391)	4.153 (0.392)	5.771 (0.294)	4.173 (0.391)	4.841 (0.269)
age	1.311 (0.439)	1.322 (0.439)	0.299 (0.448)	1.309 (0.439)	0.999 (0.285)
age^2	−2.538 (0.830)	−2.555 (0.830)	−1.099 (0.789)	−2.536 (0.830)	−2.388 (0.474)
age^3	0.117 (0.470)	0.122 (0.470)	0.145 (0.425)	0.116 (0.470)	0.882 (0.237)
$\log(lotsize)$	0.154 (0.022)	0.153 (0.022)	0.181 (0.021)	0.154 (0.022)	0.188 (0.017)
rooms	0.018 (0.020)	0.018 (0.020)	0.024 (0.014)	0.018 (0.020)	0.004 (0.010)
$\log(LTA)$	0.771 (0.061)	0.774 (0.061)	0.524 (0.040)	0.770 (0.061)	0.608 (0.037)
beds	−0.033 (0.030)	−0.032 (0.030)	−0.037 (0.020)	−0.033 (0.030)	0.019 (0.015)
syear1994	0.053 (0.046)	0.053 (0.046)	0.014 (0.030)	0.053 (0.046)	0.039 (0.024)
syear1995	0.059 (0.047)	0.059 (0.047)	0.029 (0.031)	0.058 (0.047)	0.081 (0.024)
syear1996	0.034 (0.045)	0.035 (0.045)	−0.036 (0.029)	0.034 (0.045)	0.101 (0.023)
syear1997	0.086 (0.044)	0.086 (0.044)	0.062 (0.029)	0.086 (0.044)	0.146 (0.023)
syear1998	0.155 (0.047)	0.156 (0.047)	0.119 (0.032)	0.155 (0.047)	0.194 (0.023)
ρ	0.930 (0.013)	0.926 (0.014)	0.999 (0.000)	0.930 (0.013)	0.930 (0.002)
ω	0.072 (0.006)	0.074 (0.007)	0.015 (0.000)	0.072 (0.006)	0.083 (−−)
l	- 536.778	-536.782	-887.593	-536.778	-8574.474
Iterations	1	10,000	4,614	1,290	–
			Computation Times in Seconds		
	3.5	146,268	52,091	14,822	0.8

7 Conclusion

We have reviewed various ML methods for fitting CAR models in the presence of missing data, and we found that, despite its simplicity, MML estimation considerably outperforms other iterative algorithms, including the popular EM algorithm, when efficiently implemented. We see that the iterative methods Martin84 and GBH89 are often much faster than the EM algorithm since they do not require an E-step, and we show that, as expected, Martin84 does not maximise the likelihood function. It appears that the EM algorithm is most useful when the number of missing units is small relative to n, that is, when the correlation between the latent states and the parameter estimators is low. This effect is also commonly seen in other standard statistical models, such as linear mixed models.

Acknowledgements

We would like to thank Clint Shumack for help with running the experiments on a high-performance cluster. AZM's research was supported by an Australian Research Council Discovery Early Career Research Award, DE180100203 (2018).

Supporting information

The following supporting information is available as part of the online article:
`R` (R Core Team, 2019) code for reproducing the results in Sections 5 and 6 are available at `www.uow.edu.au/~tsuesse`.

References

Bates, D. and Maechler, M. (2018). Matrix: Sparse and dense matrix classes and methods.

Besag, J. (1974). Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, 36(2):192–236.

Bivand, R. S. (2010). Comparing estimation methods for spatial econometrics techniques using R. Report, Department of Economics, Norwegian School of Economics and Business Administration. SAM 26 2010.

Bivand, R. S. (2018). spdep: Spatial dependence: Weighting schemes, statistics and models.

Bivand, R. S., Gómez-Rubio, V., and Rue, H. (2014). Approximate Bayesian inference for spatial econometrics models. *Spatial Statistics*, 9:146–165.

Bivand, R. S., Gómez-Rubio, V., and Rue, H. (2015). Spatial data analysis with R-INLA with some extensions. *Journal of Statistical Software*, 63(20):1–31.

Brent, R. (1973). *Algorithms for Minimization without Derivatives*. Prentice-Hall, Englewood Cliffs, New Jersey.

Chen, Y., Davis, T. A., Hager, W. W., and Rajamanickam, S. (2008). Algorithm 887: CHOLMOD, supernodal sparse Cholesky factorization and update/downdate. *ACM Transactions on Mathematical Software*, 35(3):22.

Cressie, N. and Wikle, C. K. (2011). *Statistics for Spatio-Temporal Data*. John Wiley & Sons, Hoboken, New Jersey.

Gómez-Rubio, V., Bivand, R. S., and Rue, H. (2017). Estimating spatial econometrics models with integrated nested Laplace approximation. *arXiv preprint arXiv:1703.01273*.

Goulard, M., Laurent, T., and Thomas-Agnan, C. (2017). About predictions in spatial autoregressive models: Optimal and almost optimal strategies. *Spatial Economic Analysis*, 12(2-3):304–325.

Griffith, D. A., Bennett, R. J., and Haining, R. P. (1989). Statistical analysis of spatial data in the presence of missing observations: a methodological guide and an application to urban census data. *Environment and Planning A*, 21(11):1511–1523.

Harrison, D. and Rubinfeld, D. L. (1978). Hedonic housing prices and the demand for clean air. *Journal of Environmental Economics and Management*, 5:81–102.

Harville, D. (1997). *Matrix Algebra From a Statistician's Perspective*. Springer, New York, New York.

Hoef, J. M. V., Hanks, E. M., and Hooten, M. B. (2017). On the relationship between conditional (CAR) and simultaneous (SAR) autoregressive models. *arXiv preprint arXiv:1710.07000*.

Kato, T. (2013). Usefulness of the information contained in the prediction sample for the spatial error model. *The Journal of Real Estate Finance and Economics*, 47(1):169–195.

Lesage, J. and Pace, R. (2004). Models for spatially dependent missing data. *Journal of Real Estate Finance and Economics*, 29(2):223–254.

Lesage, J. and Pace, R. (2009). *Introduction to Spatial Econometrics*. Chapman & Hall/CRC, Boca Raton, Florida.

Martin, R. J. (1984). Exact maximum likelihood for incomplete data from a correlated Gaussian process. *Communications in Statistics-Theory and Methods*, 13(10):1275–1288.

Ord, K. (1975). Estimation methods for models of spatial interaction. *Journal of the American Statistical Association*, 70(349):120–126.

Pace, R. and Barry, R. (1997a). Fast CARs. *Journal of Statistical Computation and Simulation*, 59(2):123–147.

Pace, R. and Barry, R. (1997b). Quick computation of spatial autoregressive estimators. *Geographical Analysis*, 29(3):232–247.

R Core Team (2019). R: A language and environment for statistical computing.

Rue, H. and Held, L. (2005). *Gaussian Markov Random Fields: Theory and Applications*. CRC Press, Boca Raton, Florida.

Suesse, T. (2018). Marginal maximum likelihood estimation of SAR models with missing data. *Computational Statistics & Data Analysis*, 120:98–110.

Suesse, T. and Zammit-Mangion, A. (2017). Computational aspects of the EM algorithm for spatial econometric models with missing data. *Journal of Statistical Computation and Simulation*, 87(9):1767–1786.

Wang, W. and Lee, L. (2013). Estimation of spatial autoregressive models with randomly missing data in the dependent variable. *Econometrics Journal*, 16:73–102.

Whittle, P. (1954). On stationary processes in the plane. *Biometrika*, 41(3-4):434–449.

APPENDIX

A Equivalence of Martin (1984)'s minimisation and concentrated log-likelihood maximisation

Martin (1984, p. 1278) suggested minimising $|\mathbf{V}_{ss}|^{\frac{1}{n_s}} \left(\hat{\mathbf{r}}_s^\top \mathbf{V}_{ss}^{-1} \hat{\mathbf{r}}_s\right)$ to fit a CAR model. This is equivalent to maximising the concentrated log likelihood (7) since

$$\begin{aligned}
&\max \left\{ -\frac{n_s}{2} \left(\log\left(\frac{2\pi}{n_s}\right) + 1 \right) - \frac{n_s}{2} \log \hat{\mathbf{r}}_s^\top \mathbf{V}_{ss}^{-1} \hat{\mathbf{r}}_s - \frac{1}{2} \log |\mathbf{V}_{ss}| \right\} \\
\Leftrightarrow &\min \left\{ +\frac{n_s}{2} \left(\log\left(\frac{2\pi}{n_s}\right) + 1 \right) + \frac{n_s}{2} \log \hat{\mathbf{r}}_s^\top \mathbf{V}_{ss}^{-1} \hat{\mathbf{r}}_s + \frac{1}{2} \log |\mathbf{V}_{ss}| \right\} \\
\Leftrightarrow &\min \left\{ \left(\log\left(\frac{2\pi}{n_s}\right) + 1 \right) + \log \hat{\mathbf{r}}_s^\top \mathbf{V}_{ss}^{-1} \hat{\mathbf{r}}_s + \frac{1}{n_s} \log |\mathbf{V}_{ss}| \right\} \\
\Leftrightarrow &\min \left\{ \hat{\mathbf{r}}_s^\top \mathbf{V}_{ss}^{-1} \hat{\mathbf{r}}_s \times |\mathbf{V}_{ss}|^{1/n_s} \right\}.
\end{aligned}$$

B Information Matrix for parameters in the CAR Model

Let $l_s \equiv -\log f(\mathbf{y}_s)$, $\mathbf{r}_s \equiv \mathbf{y}_s - \mathbf{X}_s\boldsymbol{\beta}$, and $\mathbf{B}_{ss} \equiv \mathbf{V}_{ss}^{-1} = \mathbf{M}_{ss} - \mathbf{M}_{su}\mathbf{M}_{uu}^{-1}\mathbf{M}_{us}$. It is straightforward to see that the components of the information matrix for the parameters in the CAR model are given by

$$\begin{aligned}
E\left(\frac{\partial^2 l_s}{\partial\boldsymbol{\beta}\partial\boldsymbol{\beta}^\top}\right) &= \frac{1}{\omega}\mathbf{X}_s^\top \mathbf{B}_{ss}\mathbf{X}_s, \\
E\left(\frac{\partial^2 l_s}{\partial\omega^2}\right) &= \frac{n_s}{2}\frac{1}{\omega^2}, \\
E\left(\frac{\partial^2 l_s}{\partial\rho^2}\right) &= \frac{1}{2}\mathrm{tr}\left(\mathbf{B}_{ss}^{-1}\frac{\partial\mathbf{B}_{ss}}{\partial\rho}\mathbf{B}_{ss}^{-1}\frac{\partial\mathbf{B}_{ss}}{\partial\rho}\right), \\
E\left(\frac{\partial^2 l_s}{\partial\omega\partial\boldsymbol{\beta}}\right) &= \mathbf{0}, \\
E\left(\frac{\partial^2 l_s}{\partial\rho\partial\boldsymbol{\beta}}\right) &= \mathbf{0}, \\
E\left(\frac{\partial^2 l_s}{\partial\omega\partial\rho}\right) &= -\frac{1}{2\omega}\mathrm{tr}\left(\mathbf{B}_{ss}^{-1}\frac{\partial\mathbf{B}_{ss}}{\partial\rho}\right),
\end{aligned}$$

where

$$\frac{\partial\mathbf{B}_{ss}}{\partial\rho} = \frac{\partial\mathbf{M}_{ss}}{\partial\rho} + \mathbf{M}_{su}\mathbf{M}_{uu}^{-1}\frac{\partial\mathbf{M}_{uu}}{\partial\rho}\mathbf{M}_{uu}^{-1}\mathbf{M}_{us} - \frac{\partial\mathbf{M}_{su}}{\partial\rho}\mathbf{M}_{uu}^{-1}\mathbf{M}_{us} - \mathbf{M}_{su}\mathbf{M}_{uu}^{-1}\frac{\partial\mathbf{M}_{us}}{\partial\rho}.$$

Note that some of these terms can be simplified further since $\mathbf{M} = \mathbf{I} - \rho\mathbf{W}$ and hence $\frac{\partial\mathbf{M}}{\partial\rho} = -\mathbf{W}$.